Decision Theory: Markov Decision Processes

CPSC 322 – Decision Theory 3

Textbook §12.5

Lecture Overview

Sequential decision problems

- A sequential decision problem consists of a sequence of decision variables $D_1, \ldots, D_n$.
- Each D_i has an information set of variables pD_i, whose value will be known at the time decision D_i is made.

- What should an agent do?
  - What an agent should do at any time depends on what it will do in the future.
  - What an agent does in the future depends on what it did before.

Policies

- A policy specifies what an agent should do under each circumstance.
- A policy is a sequence $\delta_1, \ldots, \delta_n$ of decision functions

$$\delta_i : dom(pD_i) \rightarrow dom(D_i).$$

  This policy means that when the agent has observed $O \in dom(pD_i)$, it will do $\delta_i(O)$.
- The expected utility of policy δ is

$$\mathbb{E}(U|\delta) = \sum_{\omega \models \delta} P(\omega)U(\omega)$$

- An optimal policy is one with the highest expected utility.

Finding the optimal policy

- Remove all variables that are not ancestors of a value node
- Create a factor for each conditional probability table and a factor for the utility.
- Sum out variables that are not parents of a decision node.
- Select a variable D that is only in a factor f with (some of) its parents.
  - this variable will be one of the decisions that is made latest
- Eliminate D by maximizing. This returns:
  - the optimal decision function for D, $\arg\max_D f$
  - a new factor to use in VE, $\max_D f$
- Repeat till there are no more decision nodes.
- Sum out the remaining random variables. Multiply the factors: this is the expected utility of the optimal policy.

Lecture Overview

Value of Information

- The value of information X for decision D is the utility of the the network with an arc from X to D minus the utility of the network without the arc.
  - The value of information is always non-negative.
  - It is positive only if the agent changes its action depending on X.
- The value of information provides a bound on how much you should be prepared to pay for a sensor. How much is a better weather forecast worth?

Value of Control

- The value of control of a variable X is the value of the network when you make X a decision variable minus the value of the network when X is a random variable.
- You need to be explicit about what information is available when you control X.
  - If you control X without observing, controlling X can be worse than observing X.
  - If you keep the parents the same, the value of control is always non-negative.

Lecture Overview

Agents as Processes

Agents carry out actions:

- forever: infinite horizon
- until some stopping criteria is met: indefinite horizon
- finite and fixed number of steps: finite horizon

Decision-theoretic Planning

What should an agent do under these different planning horizons, when

- actions can be noisy
    - the outcome of an action can't be fully predicted
    - there is a model that specifies the probabilistic outcome of actions
- the world (i.e., state) is fully observable
- the agent periodically gets rewards (and punishments) and wants to maximize its rewards received

Lecture Overview

1. Recap
2. Value of Information, Control
3. Decision Processes
4. MDPs
5. Rewards and Policies

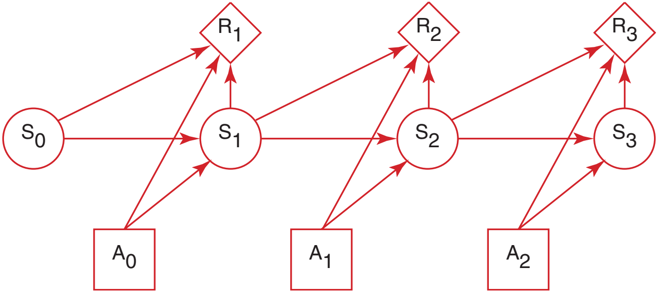
Stationary Markov chain

Start with a stationary Markov chain.

- Recall: a stationary Markov chain is when for all $t > 0$, $P(S_{t+1}|S_t) = P(S_{t+1}|S_0, \ldots, S_t)$.
- We specify $P(S_0)$ and $P(S_{t+1}|S_t)$.

Decision Processes

- A Markov decision process augments a stationary Markov chain with actions and values:



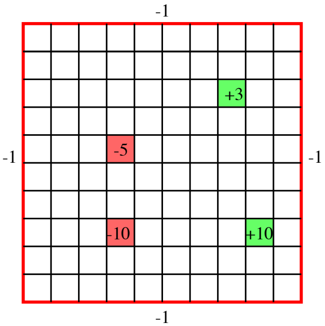
Markov Decision Processes

Definition (Markov Decision Process)

A Markov Decision Process (MDP) is a 5-tuple $\langle S, A, P, R, s_0 \rangle$, where each element is defined as follows:

- S: a set of states.
- A: a set of actions.
- $P(S_{t+1}|S_t, A_t)$: the dynamics.
- $R(S_t, A_t, S_{t+1})$: the reward. The agent gets a reward at each time step (rather than just a final reward).
  - $R(s, a, s')$ is the reward received when the agent is in state s, does action a and ends up in state s'.
- s_0: the initial state.

Example: Simple Grid World



- Actions: up, down, left, right.
- 100 states corresponding to the positions of the robot.
- Robot goes in the commanded direction with probability 0.7, and one of the other directions with probability 0.1.
- If it crashes into an outside wall, it remains in its current position and has a reward of -1.
- Four special rewarding states; the agent gets the reward when leaving.

Planning Horizons

The planning horizon is how far ahead the planner can need to look to make a decision.

- The robot gets flung to one of the corners at random after leaving a positive (+10 or +3) reward state.
  - the process never halts
  - infinite horizon
- The robot gets +10 or +3 entering the state, then it stays there getting no reward. These are absorbing states.
  - The robot will eventually reach the absorbing state.
  - indefinite horizon

Information Availability

What information is available when the agent decides what to do?

- fully-observable MDP the agent gets to observe S_t when deciding on action A_t.
- partially-observable MDP (POMDP) the agent has some noisy sensor of the state. It needs to remember its sensing and acting history.

We'll only consider (fully-observable) MDPs.

Lecture Overview

Rewards and Values

Suppose the agent receives the sequence of rewards $r_1, r_2, r_3, r_4, \ldots$. What value should be assigned?

- total reward:

$$V = \sum_{i=1}^{\infty} r_i$$

- average reward:

$$V = \lim_{n \to \infty} \frac{r_1 + \cdots + r_n}{n}$$

- discounted reward:

$$V = \sum_{i=1}^{\infty} \gamma^{i-1} r_i$$

  - γ is the discount factor, $0 \leq \gamma \leq 1$

Policies

- A stationary policy is a function:

$$\pi : S \longrightarrow A$$

  Given a state s, $\pi(s)$ specifies what action the agent who is following π will do.
- An optimal policy is one with maximum expected value
  - we'll focus on the case where value is defined as discounted reward.
- For an MDP with stationary dynamics and rewards with infinite or indefinite horizon, there is always an optimal stationary policy in this case.
- Note: this means that although the environment is random, there's no benefit for the *agent* to randomize.